

Proposed Examiner's Amendment 10/830,042 (For discussion purposes only)

7. (Currently Amended) In a multiple computer system including a plurality of single computers interconnected via a communications link, a method of loading an application program onto each of said plurality of single computers, the application program having application program code including a plurality of code threads all intended to execute on and reference a single computer having a single processing unit or symmetric multiple processing units and a single independent local memory with a local memory capacity that is not shared with any other single computer of said plurality of single computers, the method comprising the steps of:

loading the application program written to operate only on a single computer onto each different computer of said plurality of single computers; and

modifying the application program on each said different single computer before execution of said corresponding portion of the application program written to operate only on a single computer on each said different single computer;

wherein the step of modifying comprises:

(i) detecting instructions in the unmodified application program which reference the same common memory records;

(ii) listing all such commonly referenced memory records by a distributed runtime (DRT) and providing a naming tag for each said listed commonly referenced memory record;

(iii) detecting those instructions which write to, or manipulate the contents of, any of said listed commonly referenced memory records; and

(iv) generating and inserting an alert instruction into the unmodified application program to create the modified application program for handling by the DRT following each said detected commonly referenced memory record write or manipulate instruction indicating that the contents or value of the commonly referenced memory record were re-written or manipulated and may have changed during execution of a code thread, said alert instruction being operative for initiating propagation of the re-written or manipulated contents and name tag of each said re-written or manipulated listed commonly referenced memory record via the communications link to the distributed run times (DRTs) of each other of the single computers, wherein each DRT creates a table when initially recording fields, and for each field there is a name which is common throughout the network and which the network recognizes, wherein in different ones of said plurality of single computers, a memory location corresponding to a given name field will vary over time and each of the DRTs will have different memory locations but each field name will have the same field value stored in the different memory locations, and wherein the DRT initially creates a JAVA bytecode virtual machine for execution of the modified application program;

different portions of said modified application program being simultaneously executable on each different one of the plurality of single computers with each different one of the plurality of single computers having a different independent local memory accessible only by a corresponding portion of the application program.

9. (Canceled)

31. (Currently Amended) A method of compiling or modifying an application program written to include a plurality of instruction code threads intended to execute on and reference only a single computer having a single central processing unit (CPU) or symmetric multiple processing units and a single independent local memory that is not shared with any other computer of ~~said a~~ plurality of single computers, ~~said application program~~ to run simultaneously on each one of [[a]] ~~said~~ plurality of ~~single~~ computers interconnectable via a communications link, with different portions of said application program being simultaneously executable on different ones of said plurality of single computers with each one of the plurality of single computers having the independent local memory accessible only by the corresponding portion of the application program, said method comprising the steps of:

- (i) detecting instructions in the unmodified application program which reference the same common memory records;
- (ii) listing all such commonly referenced memory records ~~by a distributed runtime (DRT)~~ and providing a naming tag for each said listed commonly referenced memory record;
- (iii) detecting those instructions which write to, or manipulate the contents of, any of said listed commonly referenced memory records; and
- (iv) generating and inserting an alert instruction into the unmodified application program to create the modified application program for handling by ~~a distributed run time (DRT)~~ the DRT following each said detected commonly referenced memory record write or manipulate instruction indicating that the contents or value of the commonly

referenced memory record were re-written or manipulated and may have changed during execution of a code thread, said alert instruction being operative for initiating propagation of the re-written or manipulated contents and name tag of each said re-written or manipulated listed commonly referenced memory record via the communications link to the distributed run times (DRTs) of each other of the single computers, wherein each DRT creates a table when initially recording fields, and for each field there is a name which is common throughout the network and which the network recognizes, wherein in different ones of said plurality of single computers, a memory location corresponding to a given name field will vary over time and each of the DRTs will have different memory locations but each field name will have the same field value stored in the different memory locations, and wherein the DRT initially creates a JAVA bytecode virtual machine for execution of the modified application program.

37. (Currently Amended) A method as in claim 31, wherein the commonly referenced memory locations comprise JAVA programming language fields and the contents or values stored in the commonly referenced memory locations comprise Java JAVA programming language field contents or values.

38. (Currently Amended) A method as in claim 31, wherein the commonly referenced memory records comprise JAVA programming language fields and the JAVA programming language fields are listed by object and class.

39. (Currently Amended) A method as in claim 3 1, wherein said application program is written in the JAVA programming language and the step of detecting instructions in the unmodified application program which reference the same common memory records comprise searching through the JAVA programming language code and identifying a put static (putstatic) instruction and generating and inserting an alert instruction into the application program for each said putstatic instruction so identified.

45. (Currently Amended) In a multiple computer system including a plurality of single computers interconnectable via an Internet or intranet network communications link, a method of loading an original application program onto each of said plurality of single computers, the original application program having original application program code including a plurality of original code threads all written to execute on and reference a single computer having a single processing unit or symmetric multiple processing units and a single local memory with a local memory capacity that is not shared with any other single computer of said plurality of single computers, the system configured to enable simultaneous cooperative execution of said application program by said plurality of single computers, with the original application program being modified to form at least one modified application program with different portions of said modified application program being simultaneously executed within a different independent local processor and a different independent local memory within each different one of the plurality of single computers, said different independent local memory within each said different single computer not forming a distributed shared memory arrangement and being accessible during execution of said application program and said different portions of

said application program only by the different portion of the application program actually executing within the different local processing unit or symmetric multiple processing units of the different computer, the method comprising the steps of:

loading the application program onto each different computer of said plurality of single computers, said application program including a reference to a program memory field that may be referenees referenced by one or more of said plurality of computers during execution of their respective different portion of the application program; and

modifying the application program on each said different single computer before execution of said different portion of the application program on each said different single computer; wherein the step of modifying comprises:

- (i) detecting instructions in the unmodified application program which reference the same common memory records;
- (ii) listing all such commonly referenced memory records by a distributed runtime (DRT) and providing a naming tag for each said listed commonly referenced memory record;
- (iii) detecting those instructions which write to, or manipulate the contents of, any of said listed commonly referenced memory records; and
- (iv) generating and inserting an alert instruction into the unmodified application program to create the modified application program for handling by the DRT following each said detected commonly referenced memory record write or manipulate instruction indicating that the contents or value of the commonly referenced memory record were re-written or manipulated and may have changed during execution of a code thread, said alert instruction being operative for

initiating propagation of the re-written or manipulated contents and name tag of
each said re-written or manipulated listed commonly referenced memory record
via the communications link to the distributed run times (DRTs) of each other of
the single computers, wherein each DRT creates a table when initially recording
fields, and for each field there is a name which is common throughout the
network and which the network recognizes, wherein in different ones of said
plurality of single computers, a memory location corresponding to a given name
field will vary over time and each of the DRTs will have different memory
locations but each field name will have the same field value stored in the different
memory locations, and wherein the DRT initially creates a JAVA bytecode virtual
machine for execution of the modified application program; and
said modification of the application program includes an insertion of at least one
code thread prior to execution that upon execution by one of said single computers
initiates a sequence of events that result in a network packet communication over said
Internet or intranet network communications link that contains an identifier of the
referenced memory field and the contents or value of that memory field.